A Primer to Subject-Oriented Business Process Modeling

Albert Fleischmann[1], Werner Schmidt[2], and Christian Stary[3]

[1] Metasonic AG, Münchner Straße 29, Hettenshausen, 85276 Pfaffenhofen, Germany
Albert.Fleischmann@Metasonic.de
[2] Applied Sciences Ingolstadt, Esplanade 10, 85049 Ingolstadt, Germany
Werner.Schmidt@haw-ingolstadt.de
[3] Johannes Kepler University Linz, Freistaedterstrasse 315, 4040 Linz, Austria
christian.stary@jku.at

Abstract. This primer aims towards quick wins in BPM based on a conceptual understanding of S-BPM modeling. We put forward the main constructs in terms of model elements and diagrams. Addressing primarily non-S-BPM-experts we introduce fundamental approaches, namely S-BPM modeling by construction (i.e., when starting from scratch), and S-BPM modeling by restriction (i.e., when refining a generalized peer-to-peer network). Since S-BPM modeling is closely related to standard sentence semantics we start out revisiting essential natural language constructs (subject, predicate, and object) and their role in communication patterns between actors (subjects). We then show how S-BPM constructs can be organized and presented, and introduce interaction and behavior diagrams. Their application ensures coherent representations, and facilitates creating intelligible specifications when business processes are captured from a stakeholder perspective. Finally, we show the essential role S-BPM models play along the open S-BPM life cycle, ranging from process analysis to execution and monitoring.

Keywords: business process modeling, subject orientation, model construction.

1 Introduction

Process specifications play a crucial role in S-BPM [4]. Focussing on the stakeholder perspective when identifying and specifying processes requires a switch from a purely function-driven to an actor- or a system-driven perspective. In particular, if stakeholders should be able to design their organization in a self-organized, however informed way, they need to build up modeling capacities with respect to participatory organization development [4].

In S-BPM, both, the intuitiveness of the notation, and the coherence of the approach to modeling and processing, reduces the semantic distance between workflow processing and human understanding of business processes. It should require minimal cognitive effort for all stakeholders to recognize and communicate subsequently modeled business processes. Once stakeholders are able to describe and (re)present their organization of work in such a self-contained way, they may share

C. Stary (Ed.): S-BPM ONE 2012, LNBIP 104, pp. 218–240, 2012.

and communicate organizational designs - a prerequisite to participatory and dynamic development of organizations [1] [4] [5].

When describing business processes stakeholders may either start from scratch or (re-)use existing process models. We will detail both ways, as each of the approaches has certain implications for follow-up activities in the open S-BPM lifecycle. In any case, S-BPM allows stakeholders utilizing natural language structures the same way they compile standard sentences. They should be able to map knowledge about business processes (expressed in terms of standard sentence semantics) to S-BPM diagrams with minimal additional cognitive effort.

All diagrammatic S-BPM representations focus on the interaction among actors or systems, denoted as subjects. The approach allows refining interaction patterns of subjects in terms of exchanging messages, both on a general level of description, and on the level of executable specifications. The latter enables hands-on experience of validated S-BPM models when using a corresponding S-BPM suite, such as Metasonic (www.metasonic.de)

In section 2 we motivate the role of natural language and stakeholder-oriented communication structures. We also describe fundamental concepts to create S-BPM models. Section 3 introduces S-BPM modeling by construction. Hereby, modelers start from scratch identifying involved stakeholders (subjects) and their behavior in terms of sending and receiving messages (i.e. describing mail connections), in addition to individually performed activities (functions). Starting from scratch is a common approach to modeling in BPM - cf. [7].

In section 4 we revisit modeling by restriction, starting out with predefined general interaction patterns between subjects (i.e. actors and/or systems). After giving concrete names to each of the involved subjects modelers need only to keep those communication links that are required for effective task accomplishment. All others are removed from a diagrammatic S-BPM scheme [8].

Modeling by restriction is likely to reduce development efforts, as it starts with a generic, while automatically executable behavior pattern. Hence, the reduction of interaction relations does not hinder the seamless execution of S-BPM models [8].

In section 5 we explain the various action bundles in S-BPM, i.e. the open S-BPM lifecycle, and the role subject-oriented models play in each of the bundles. Section 6 concludes the primer summarizing the main elements for S-BPM capacity building.

2 Approaches to S-BPM Modeling

This section motivates and details the use of essential natural language constructs for the representation of business processes in S-BPM. Modelers may either start modeling from scratch through step-by-step construction, or from generic interaction patterns by restricting interactions according the organization of work refining and instantiating a general network of actors or systems.

S-BPM originates from the observation that humans, when structuring and describing their observed reality, use subjects, predicates, and objects. Each of them can be mapped to natural language entities. They support human communication effectively, both in written and oral form. In addition, humans use natural language structures as primary means to ensure mutual understanding [6]. In S-BPM we make use of it, as it facilitates understanding business process models, and sharing of these models.

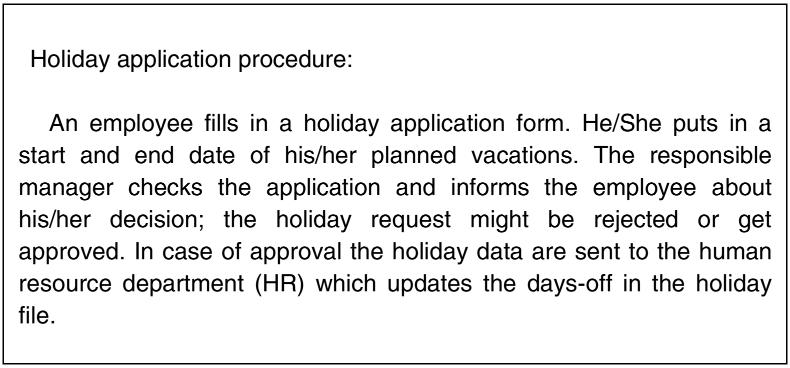
The S-BPM modeling language captures the above mentioned constituent elements of natural language sentences. Models describe structural properties and behavioral alternatives, including the interaction occurring in the technical and/or organizational environment. S-BPM models can be transformed step by step into an executable application in a seamless way. In order to ensure coherence of specifications, the exchange of messages determines the flow of control. As such, S-BPM enriches flow concepts of function-driven BPM approaches by actors sending and receiving mails.

Modeling means to represent parts of the observed reality in terms of languages. In case of S-BPM natural language terms are used, as they allow for universal use and are familiar to stakeholders through daily communication. S-BPM uses the standard semantics for sentences, comprising subject, predicate and object:

- A *subject* is the starting point for describing a situation or a sequence of events.
- Activities are denoted by *predicates.*
- An *object* is the target of an activity.

Existing modeling approaches tend to focus on predicates or objects, finally adding the subject for natural language explanations of the represented information (cf. identifying function trees before specifying eEPCs in ARIS [7]). In contrast to these approaches S-BPM modeling is a 'subject-first' approach. In further steps the functional behavior of subjects, including task-relevant communication and object (data) exchange, is described. For a more detailed discussion of S-BPM in the context of traditional approaches see [3, pp. 315 ff.].

Models address both, individual work tasks, and organization-wide ones. In the course of accomplishing their tasks, stakeholders receive work inputs, process them, and pass on results. Hence, interaction and communication, either direct or indirect, are to be considered as an essential activity of stakeholders or (IT-)systems for subject-oriented modeling.



Holiday application procedure:

An employee fills in a holiday application form. He/She puts in a start and end date of his/her planned vacations. The responsible manager checks the application and informs the employee about his/her decision; the holiday request might be rejected or get approved. In case of approval the holiday data are sent to the human resource department (HR) which updates the days-off in the holiday file.

Fig. 1. Natural language description of an application process for holidays

We exemplify S-BPM modeling using a common scenario in organizations. Employees have to apply for going on holidays or taking some days off. Figure 1 shows the natural language description of the respective process.

This simple holiday process can be modeled following two different approaches (see for details section 3 and 4, respectively). They differ in the starting point specifying a process. The traditional approach (modeling through construction) starts from scratch (‘empty sheet’-approach), whereby the process model is constructed step by step, subject-by-subject. Task-relevant actors or systems need to be identified as the process specification evolves, and the lines of interaction need to be included as required for each subject's task accomplishment. In each step, the process description becomes more complete with respect to work results.

The other approach - modeling through restriction - is only available in S-BPM. It starts with a generic process model which is restricted step by step. When beginning to model, it is assumed that all involved actors or systems might interact mutually. Hence, they can be predefined in a networked structure. In the course of modeling the lines of interaction (i.e. communication relationships) need to be adapted to those required for task accomplishment.

Figure 2 shows both approaches while sketching the conceptual difference between the restrictive and the constructive approach to modeling.

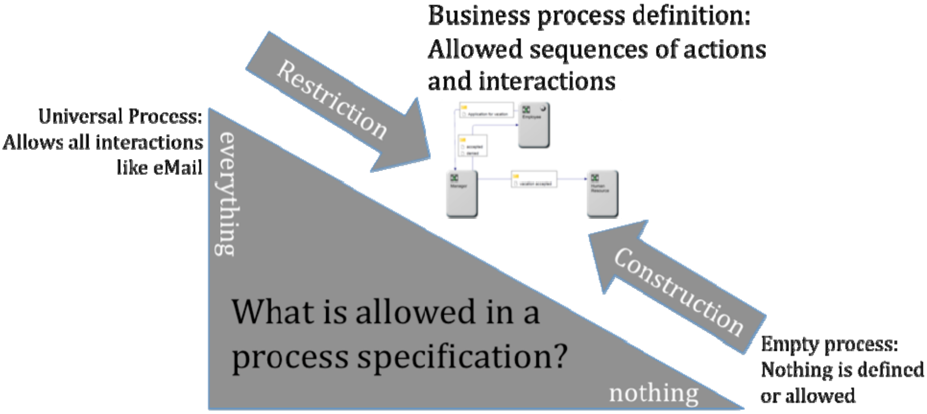

Business process definition:
Allowed sequences of actions and interactions
Restriction
Universal Process:
Allows all interactions like eMail
everything
Application for vacation
Employee
accepted
denied
Manager
vacation accepted
Human Resource
Construction
What is allowed in a process specification?
nothing
Empty process:
Nothing is defined or allowed

Fig. 2. Approaches to model business processes in S-BPM

In the following both approaches will be explained in detail. In section 3 the stepwise creation of a process model is detailed. In section 4 the stepwise reduction of interactions between actors or systems is explained. In both cases actual or envisioned work processes can be represented in a transparent and traceable way. In case validation effort for directly generating workflows from S-BPM models should be minimized, modeling by restriction provides some benefits.

3 Modeling through Construction

In this section, the 'empty-sheet' approach to identify and specify business processes in S-BPM is explained. After giving the steps to follow we introduce the S-BPM subject and interaction representations.

3.1 Procedure to Follow

Subject-oriented modeling of processes applying the construction approach comprises the following major activities:

- identification and description of the subjects involved in the process, as they occur in the course of accomplishing tasks,
- identification and description of interactions the subjects are part of,
- specifying the messages they send or receive through each interaction, and
- detailing the behavior of each subject.

3.2 Subjects and Their Interactions

As subjects are abstract resources representing the parties involved in a process the modeling process starts with identifying the involved subjects and their interactions on a high level of description. It is continued by defining the behavior specifications of each acting party. For completion all exchanges of messages required for achieving work results have to be specified. Each subject is directly addressed, as perceived in reality.

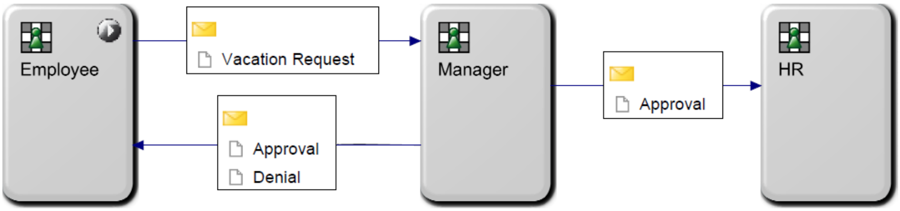


Fig. 3. Identified subjects and their communication - the Subject Interaction Diagram

Figure 3 exemplifies the identified subjects and the messages they exchange for the holiday application procedure explained in Figure 1. The modeler has identified the following subjects:

- Employee
- Manager
- Human Resource Department (HR)

The messages they need to exchange according to the scenario description in Figure 1 are:

- Vacation Request from Employee to Manager
- Approval or Denial from Manager to Employee
- Approval from Manager to Human Resource Department (HR)

The resulting diagram is termed Subject Interaction Diagram as it contains all the subjects involved and the interaction relations they need to have for accomplishing a certain task.

3.3 Subject Behavior, States and State Transitions

The behaviour of subjects is described by three states (send, receive, internal function) and transitions between these states. These states represent predicates (operations), which means, that they are active elements of the subject description. Services are being used to implement the states. State transitions are necessary to exchange and manipulate business objects.

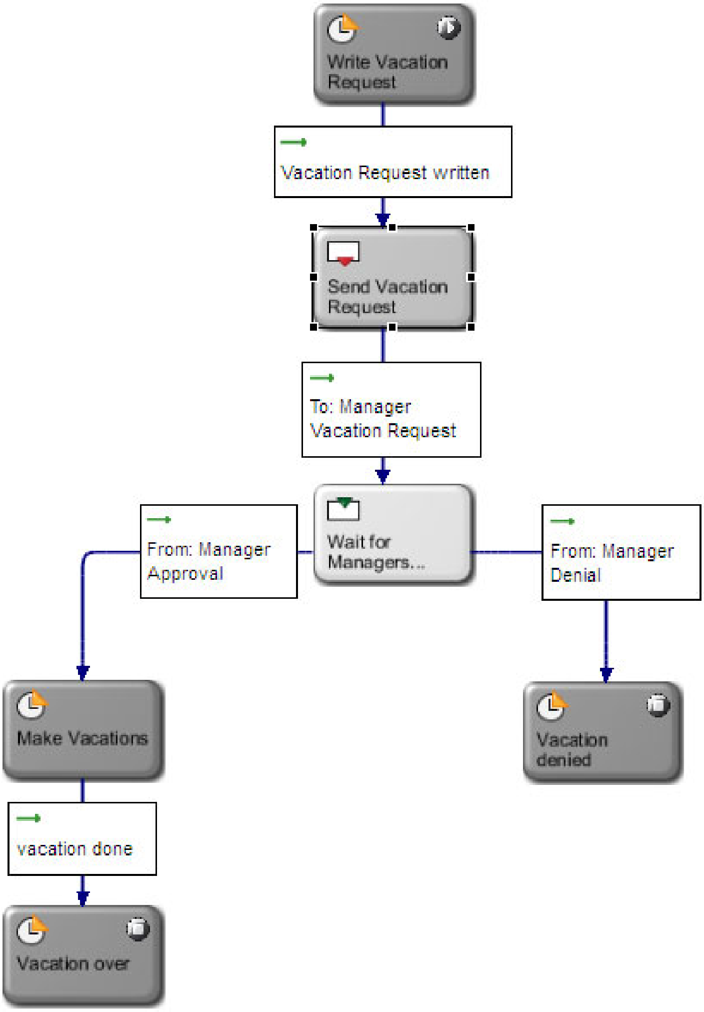


Fig. 4. The S-BPM Subject Behavior Diagram for Employee

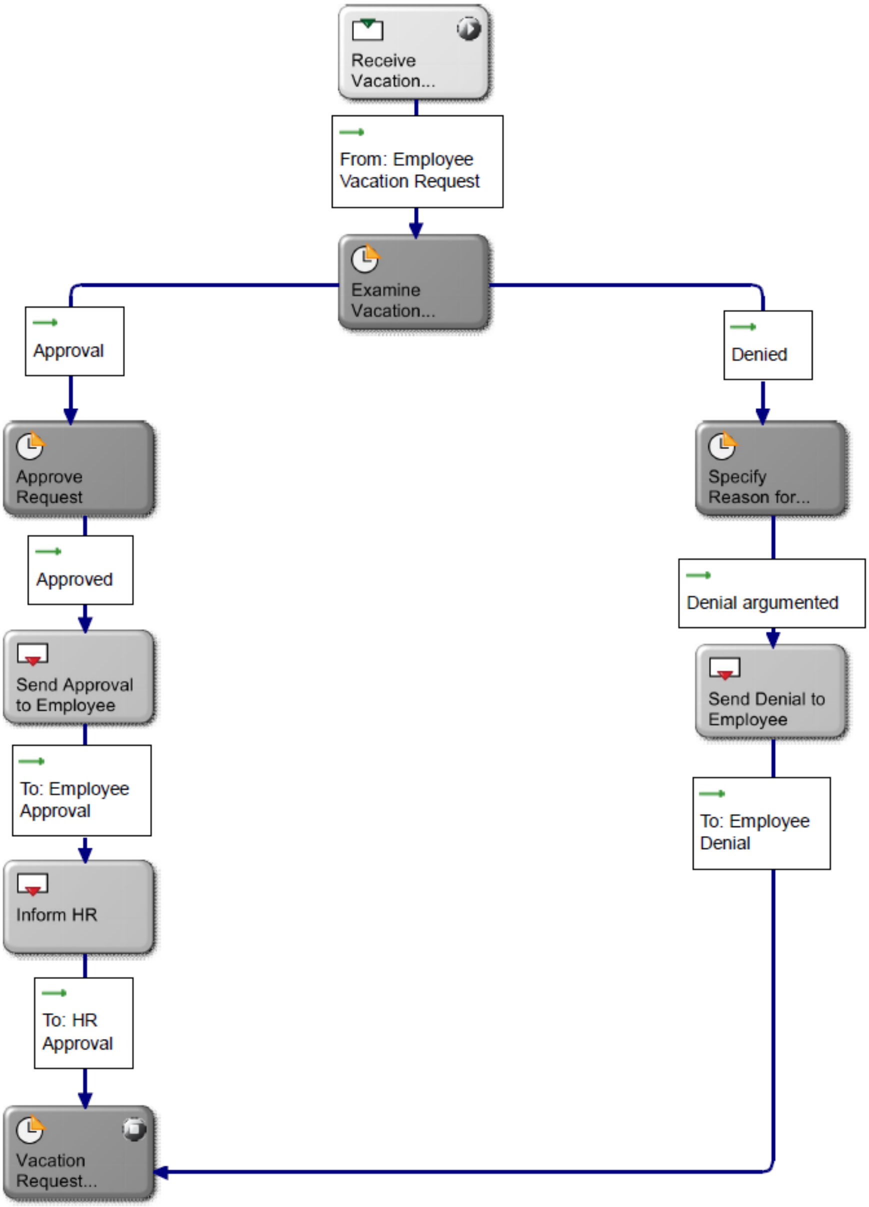


Fig. 5. Subject Behavior Diagram for subject Manager in holiday application process

When specifying the behavior of each subject, as shown in Figure 4 for the employee, a sequence of sending and receiving messages, and activities to be set for task accomplishment need to be represented. The initial state on top is marked by a 'Play' symbol in the upper right corner. In this state the employee fills in a holiday

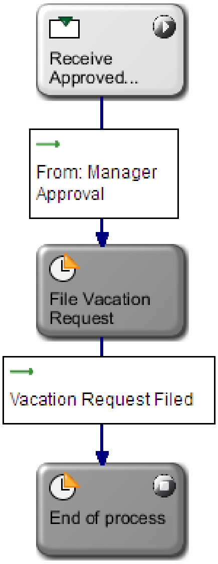


Fig. 6. Subject Behavior Diagram for HR department behavior in holiday application process

application form. Upon completion the employee's state switches to the next state via the transition 'Vacation Request written'. This state is a sending state. In this state the holiday application is sent to the manager.

After successful sending the employee reaches the state 'Wait for manager's answer' waiting for approval or denial. This state is a receiving state. In case of denial the process terminates ('Stop' symbol in the upper right corner). In case of approval, the holidays can be taken as applied for. Upon return of the employee the holiday application process terminates, too.

In Figure 4 the employee behavior in the holiday application process is encoded as follows:

- dark grey = function state, e.g., Write Vacation Request
- medium grey = send state, e.g., Send Vacation Request
- light grey = receive state, e.g., Wait for Manager's Answer
- white = state transition, e.g., Vacation Request Written
- start state = on top, i.e. Write Vacation Request
- end state = in the end of sequence of states and transitions, e.g., Vacation Over

The behavior of the manager is complementary to the employee's. The messages sent by employee are received by the manager and vice versa.

Figure 5 shows the behavior of the manager. The manager is on hold for the holiday application of the employee. Upon receipt of the Vacation Request the holiday application is examined (function state). This check can either result in an approval or a denial, leading to either state, informing the employee, and HR (only in case of approval).

In case the holiday application is approved, the HR department is informed about the successful application, and for the subject Manager the process comes to an end.

Finally, the behavior of the HR department has to be detailed (see Figure 6). HR receives the approved holiday application and puts it to the employee's days-off record, without further activities (process completion).

3.4 Services

The description of a subject defines the sequence of sending and receiving messages, or the processing of internal functions, respectively. In this way, a subject specification contains a task-relevant sequence of predicates. Predicates can be of the type 'send', 'receive' or 'internal function', the latter dealing with specific objects, such as required when an employee files a holiday application form.

As a consequence at least one operation needs to be assigned to each state. Further detailing of operations is not necessary at the modeling stage of S-BPM (see also lifecycle in section 5), as operations might be processed by existing applications. For instance, filling in a Vacation Request could be supported by a transaction of an ERP (Enterprise Resource Planning) system. A corresponding form based on the structure of an employee data record could be processed for application purposes.

Figure 7 shows how the predicates of a subject are defined by means of objects. They encapsulate all relevant data manipulations based on the Subject Behavior Diagram. Hence, the business object Vacation Request Form for the Holiday Application case contains the following operations:

- Examine application
- Approve Request
- Specify Reason for Denial
- Vacation Finished - Inform HR

As we abstract from implementation details, it is suitable to replace the term operation by the more general term service. A service is assigned to a state and thus, is triggered and processed once the state is reached. The name of the states and the names of the assigned services can be different, as shown in Figure 7. Such differences indicate that in a state several services can be used, in order to define the required functionality executed in a state.

The end conditions correspond to links leaving the state. Each result link of a sending state is assigned to a named service. Before sending this service is triggered to identify the content or parameters of a message. The service determines the values of the message parameters transferred by the message.

Analogously, each output link of a receiving state is assigned to a named service. When accepting a message in this state that service is triggered to identify the parameters of the received message. The service determines the values of the parameters transferred by the message and provides them for further processing.

All services are triggered in a synchronous way, i.e., a subject only reaches its subsequent state once all services called in a certain state have been completed.

3.5 Business Objects

For a more complete understanding of the process the content transferred by the send and receive services, and the data processed by internal function services need to be refined. Using the construct 'business object' modelers can define data structures describing the content of messages. They capture process-relevant data elements (with attributes like type, value domain, default values) and nested data structures [3].

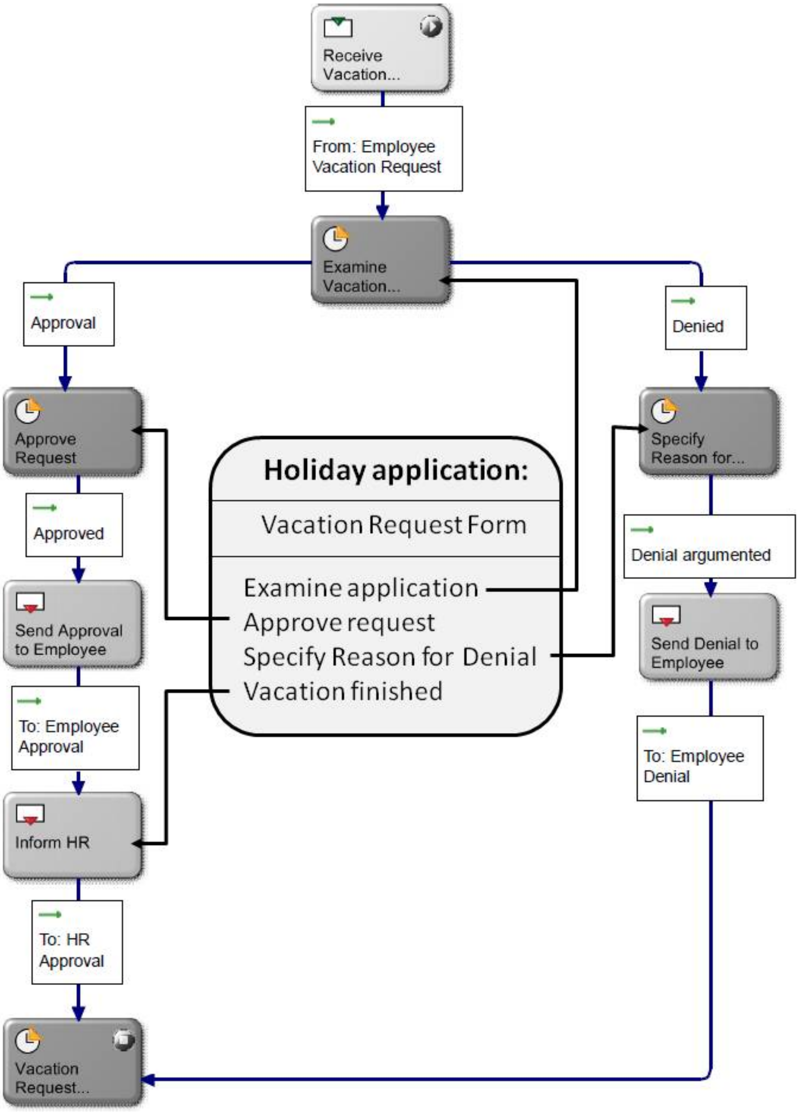


Fig. 7. Subject with predicates and corresponding object

Modeling different states of business objects allows physically adding or deleting data elements during runtime. This facilitates process design, e.g., when there is the need restricting the flow of data to business partners across enterprise borders. In addition, defining views on business objects allows limiting the access of subjects to data that exists physically in object instances.

For our example the business object Vacation Request Form is relevant to run the process. In the state Employee it contains data such as internal number, name of the employee, number of days off already taken and remaining, and start and end date of the vacation period requested. Some of these pieces of information are filled in manually by the applicant, others automatically by a service connected to the HR database. In the state Manager a data element capturing absent days due to sickness is available additionally. It could be filled in by a service, and displayed once the manager receives the request for decision making.

4 Modeling through Restriction

In this section, the generic networked approach to specify business processes in S-BPM is explained. After giving the steps to follow we introduce the generic network of interlinked subjects and exemplify its stepwise adaptation for the holiday application case. Finally, we sketch some practical benefits when following this approach in BPM.

4.1 Procedure to Follow

As mentioned in section 2 the restriction approach in S-BPM starts with an overall generic process model. The procedure requires several restriction steps:

1. Specify a generic template according to the number of parties involved in handling a certain business case (cf. Figure 8)
2. Name the subjects according to the application domain
3. Identify and remove message connections between subjects which are not necessary
4. Name messages and introduce message types according to the application (domain)
5. Adapt specification to actual subject behavior
6. Refine the structure of the business objects transmitted by the various messages

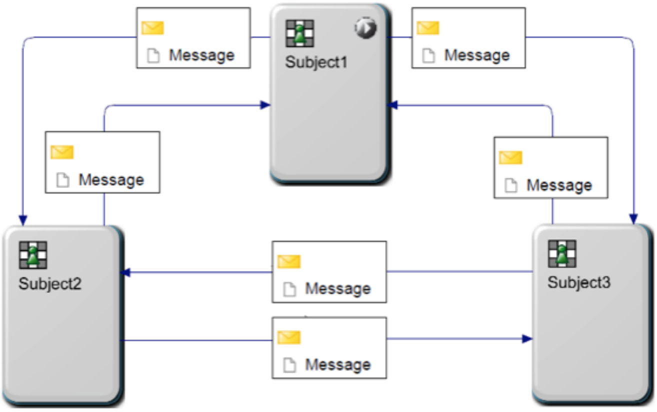


Fig. 8. Subject-oriented representation scheme for a 3-party process

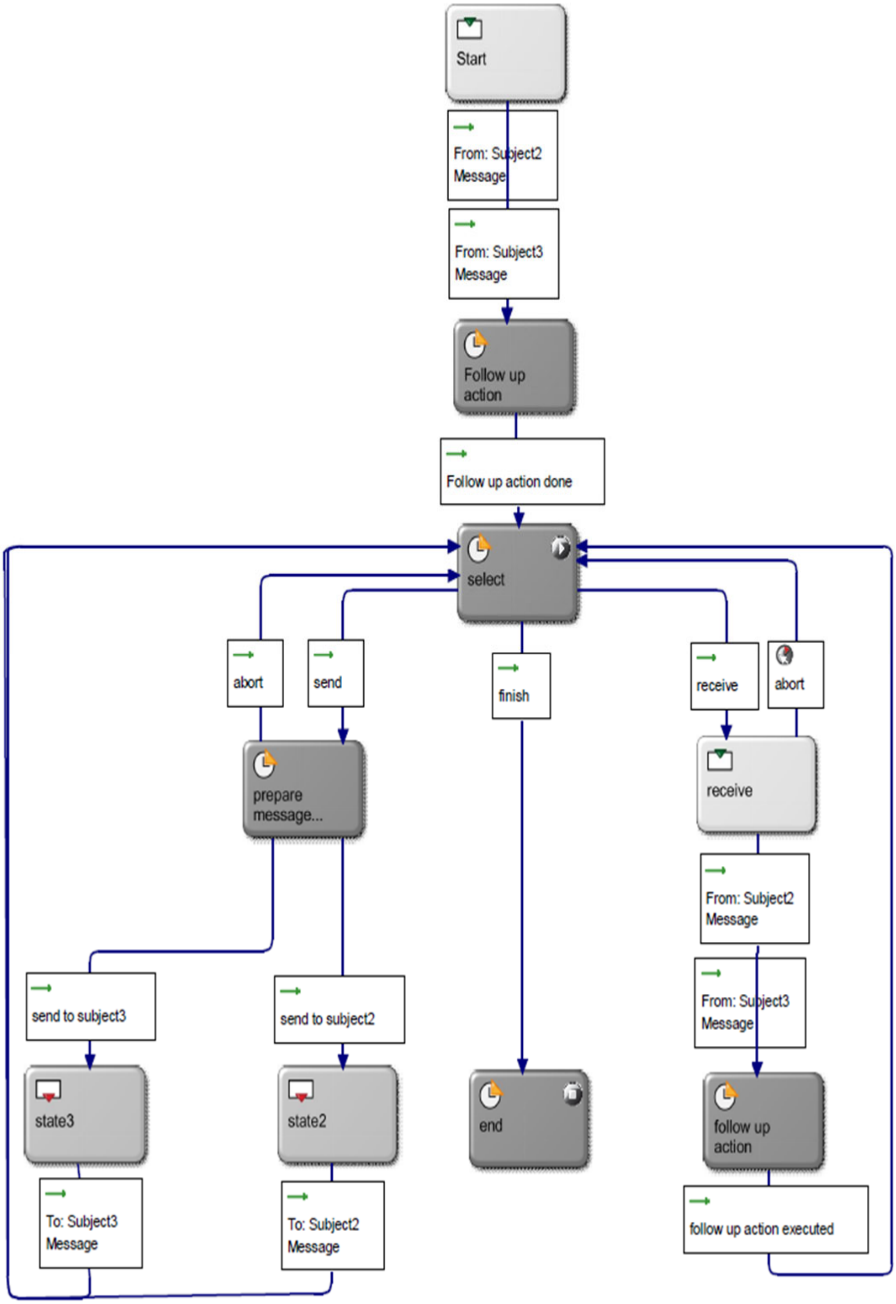


Fig. 9. Generic behavior of the start subject Subject1

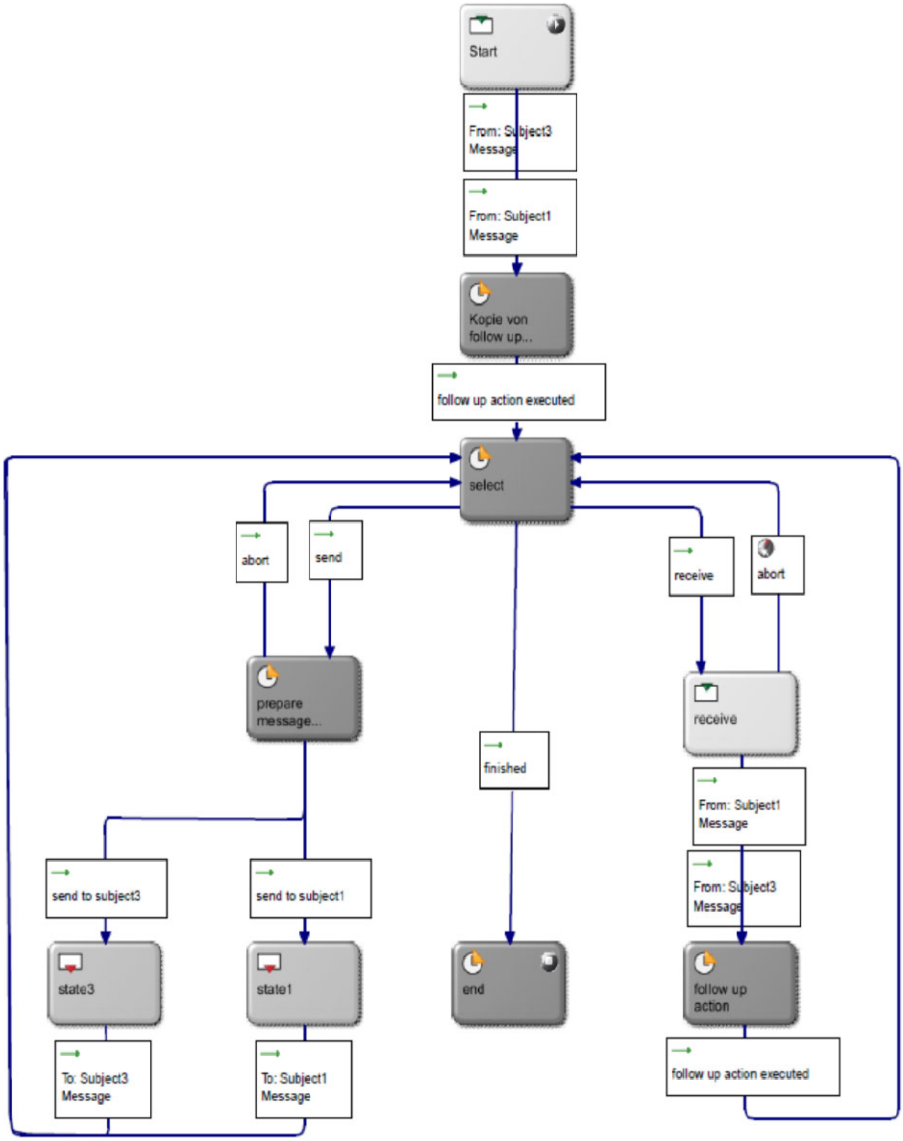


Fig. 10. Generic behavior of Subject2

4.2 Set Up of Generic Specification Scheme

In the first step a generic template according to the number of parties involved in handling a certain business case needs to be specified. The following figure shows a generic subject-oriented specification scheme with 3 involved parties. It fits to the holiday application process, as the 3 subjects required for are, according to the scenario described in Figure 1: employee, HR department, and manager. In principle, each of the parties can exchange messages with another party. We want to show how this generic process specification can be restricted step by step, in order to achieve a process specification representing the holiday process described in Figure 1.

Each subject starting message exchange is marked with a ‘Play’ symbol (small white triangle) in the upper right corner, as in Figure 8 Subject1.

In principle, each subject can send messages with the name ‘Message’ to any other subject any time. Figure 9 shows the behavior of the subject with the name ‘Subject1’. Since Subject1 is the subject which starts a process its start state is the state ‘select’. The start state is marked with a ‘Play’ symbol. The state ‘start’ and the transitions to the state ‘select’ will be never executed in the start subject.

In the behavior specifications of all other subjects the 'start' state is a 'receive' state because they are all waiting for a message of any other subject (see Figure 10).

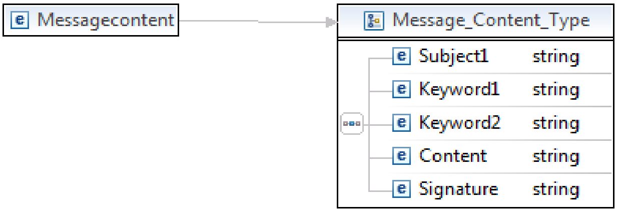
In this way all subjects that are not start subjects have to receive at least one message before they can start to send messages. The start subject sends a message to any other subject. The receiving subject can now reach the state ‘select’. In that state any subject can decide upon its next action without restriction. A subject which is in state ‘select’ can send a message to other subjects which are still in the state ‘start’. Now these subjects can also reach the select state and can send messages. Finally, all subjects are in the state ‘select’ and can communicate when addressed.

In the 'select' state the start subject decides whether it wants to send or to receive a message. To start a workflow it does not make sense to receive a message because all the other subjects are waiting for messages (as mentioned before their start state is a 'receive' state). Consequently, the start subject will start with sending messages and the exchange of messages can begin. Choosing the 'send' transition the subject moves to the state ‘prepare message and select address’ and fills out the business object (i.e., the data to be manipulated in the course of task accomplishment). That business object is transmitted by the message ‘message’. After that the subject decides to which other subject the message with the business object as content will be sent.

In the 'select' state a subject can also decide whether it wants to receive a message - this choice can make sense for a start subject further on when moving into the 'select' state the second time.

If there is a message for the subject available it can be accepted and a follow up action can be executed. It is not specified what the follow up action is. This is like receiving an e-mail. The receiver can interpret the content of an e-mail and knows what the corresponding follow up action is. The abort transitions back to the select state enable to step back in case a subject has made the wrong choice.

The representation scheme can easily be set up for any number of participants, following the same principles as shown for 3 parties. The behavior of each subject has to be adapted to the number of subjects in a process. In the send area transitions must be added to send a message to every single new subject, and the same is necessary for the receive area. With that extension scheme the behavior for each type of multi-party process can be generated automatically.



Subject1	string
Keyword1	string
Keyword2	string
Content	string
Signature	string

Fig. 11. Generic structure of the 'Message' Business Object

With the message ‘Message’ a business object is sent. The structure of this business object corresponds to the structure of a traditional e-mail with extensions like subject (attention: Here the word ‘subject’ has a different meaning. It can mean topic, issue, theme etc.), keywords and signature. Figure 11 shows the specification of the business object ‘Message’ in XSD notation (XML Schema Definition).

4.3 Adaption of Generic Scheme to a Specific Application Domain

Following the modeling steps in section 4.1 a process specification is developed corresponding to the involved parties in a business process and their generic interaction structure. In our example the restriction steps (omitting all interactions not relevant for holiday applications) result in a communication structure shown in Figure 12, and a behavior specification of the subject ‘Employee’ shown in Figure 13.

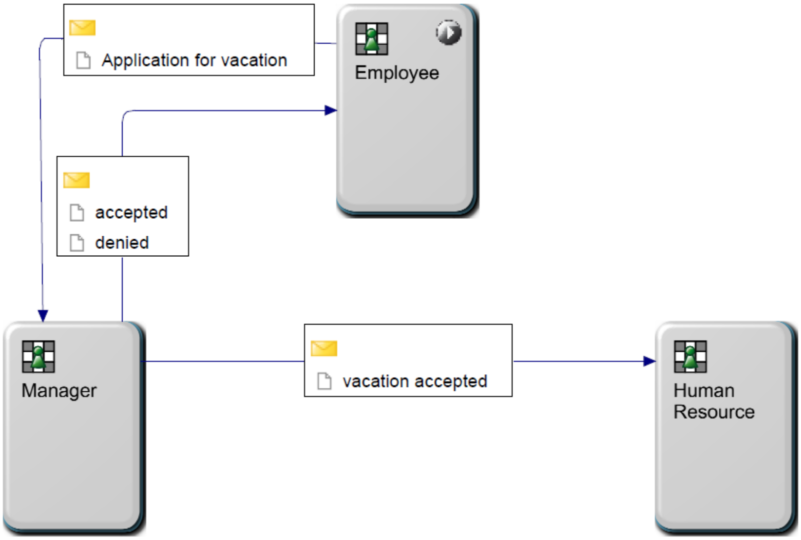


Fig. 12. Subjects and exchanged messages after restricting the interaction

With each step in restricting communication, a work task for subject holders becomes more transparent with respect to required inputs for task completion and results. In this way, S-BPM guides organizational developers starting with a network of mutually communicating stakeholders and achieving a focused, since actually required communication scheme for accomplishing tasks.

Comparing Figure 13 with Figure 4 shows that modelling through restriction does not necessarily result in models identical to those created by modeling through construction. Nevertheless both models must deliver the requested results.

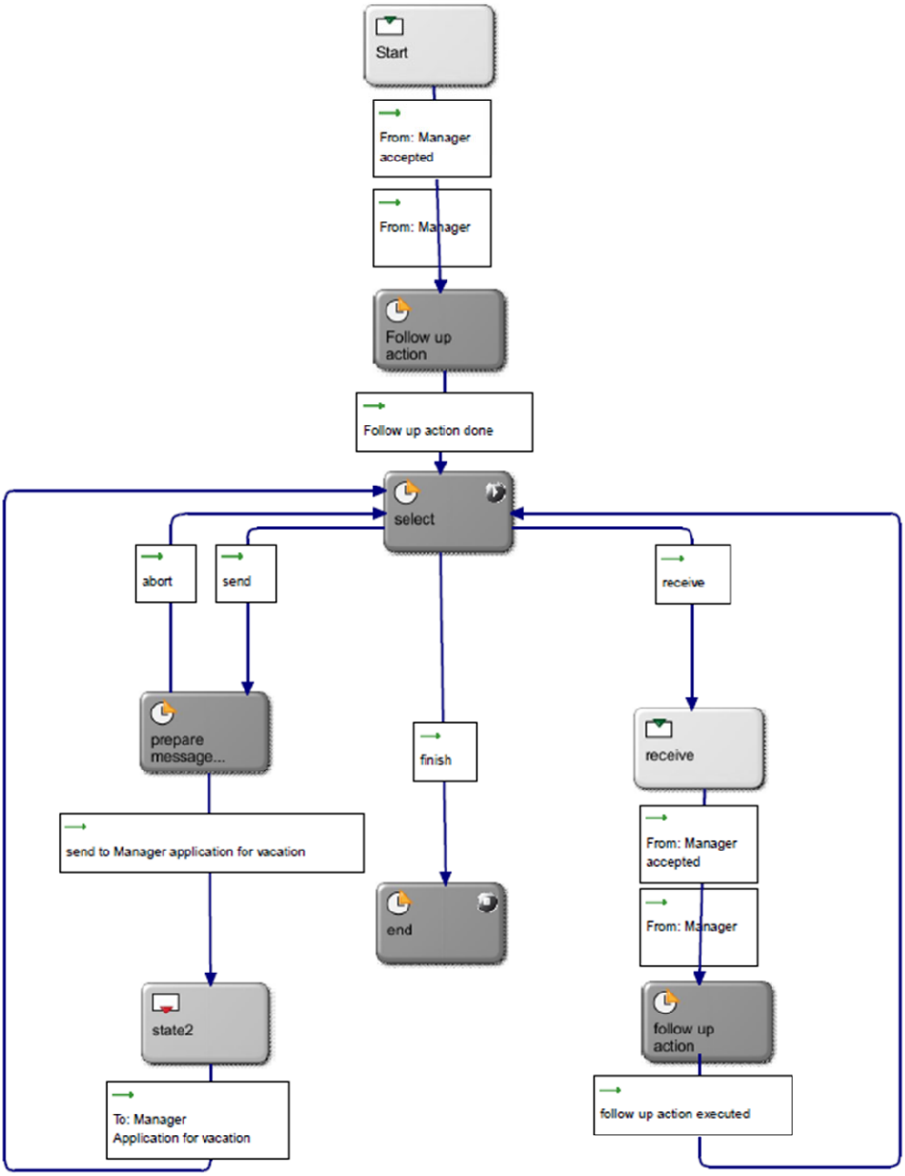


Fig. 13. Concrete behavior of subject ‘Employee’

4.4 Some Practical Benefits of Modeling through Restriction

The benefits of modeling through restriction can be best explained when comparing it with modeling through construction. As the latter corresponds to a blank paper approach, modeling by restriction is the complete opposite. Modeling by construction starts from scratch, as subjects and their behavior are modeled, once they are considered to be relevant by individual modelers or stakeholders. Subjects and communication relationships are specified in a cumulative way. Communications patterns are defined and explored as the respective modelers or stakeholders perceive work procedures. Each model develops over time and represents the current state of business affairs at a certain point in time. It is not linked to a baseline, such as the generic frame of reference for modeling through restriction, in order to minimize redundancy or provide a certain structure for design cycles. Consequently, revisiting S-BPM models might cause additional modeling workload.

When modeling through restriction modelers utilize a structural frame of reference, as the various possibilities for subjects to interact with each other can be predefined, once the number of involved parties in a business process has been identified. Such a baseline aims to minimize the modeling workload, due to the completeness of the communication pattern set up in the beginning of the modeling process - all possible interactions (i.e., message relationships) between subjects are represented (cf. Figure 8). The generic message relationships serve as placeholders which are removed in case they are not required for completing the process at hand (i.e., as soon as they cannot be named according to their task-specific purpose).

In case of revisiting S-BPM models the generic frame of reference can be revoked to facilitate checking the completeness of an S-BPM model. Candidates for further modeling (either for removal or concrete naming) can be identified easily, as those elements still carry generic labels, such as Message.

Finally, modeling by restriction facilitates the automated execution of S-BPM models. The generalized peer-to-peer network (frame of reference) contains all the subjects that are relevant for a business operation at hand. Since it also contains the possible communication relationships between the subjects, this model represents an S-BPM Interaction Diagram (cf. Figure 8). It contains a complete control flow description for generating workflows. Using a corresponding interpreter (cf. [3]) S-BPM models can be executed on demand - business processes can be experienced interactively, even when some subjects and messages have not been assigned to concrete actors, systems and message paths.

Overall, modeling by restriction is likely to reduce S-BPM efforts, as it starts with a generic, while automatically executable behavior pattern. Neither the reduction of message paths nor level of abstraction hinders the immediate execution of S-BPM models when modeling through restriction. Such models can rather be embedded without further transformations in action bundles of the open S-BPM lifecycle dealing with workflow generation and monitoring (see section 5).

5 Models and the Open S-BPM Lifecycle

S-BPM allows organizations to be developed with respect to business functionality and technical process support (cf. [2]). Business-specific aspects of BPM concern relevant management activities such as documentation, design, implementation, control and further development of management processes. They need to be organized in a way that organizations can become fully aligned to the behavior of its stakeholders and its IT support capabilities.

The S-BPM lifecycle provides an open structure for organizing S-BPM activities, as explained in the following subsections. First we detail the different roles required to trigger, guide, implement, and reflect S-BPM projects. Then, we explain the various bundles of activities that can, but need not to be performed in a linear sequence when recognizing the peculiarities of the S-BPM approach. The latter determine the central role of S-BPM models for dynamic organizational development driven by stakeholders, as discussed in the final subsection.

5.1 S-BPM Roles

(S-)BPM activities are driven and performed by persons acting in certain roles. Although each of them needs to be considered in the context of the bundle of activities of the open S-BPM lifecycle (see section 5.2), they can be characterized in general as follows:

- **Governors.** They take care about the constraints under which BPM activities are performed. They focus on influential factors that are relevant for change processes, such as market forces or structural particularities of the organization at hand. The governor's tasks range from strategic to operational development. As such, governors address all BPM-relevant stakeholders with respect to organizational development issues. However, they are not responsible for content-wise development and the domain-specific procedures that drive the value chain of a business and are executed by the actors.
- **Actors.** They execute business procedures. Hereby, they manipulate business objects and interact mutually, in order to deliver products and services. They are supported by experts and facilitators with respect to S-BPM activities.
- **Experts.** They are IT-architects, organizational developers, or specific domain specialists. They become part of S-BPM activities once their expertise is required. Typical examples are data engineers, embodying business objects into data management facilities of an organization.
- **Facilitators.** They guide the development process. They handle inputs to (S-)BPM, while ensuring the social acceptability of change proposals on the organizational level. In their interventions they tackle the method and social dimension of organizational change processes. They serve as moderators, and, if required, as mediators.

Governors, actors, experts, and facilitators are involved in each of the bundles of activities (see section 5.2). They are required to analyze business constraints when operating processes, to model and bring models to live, and to learn from and through changes. Their mutual cooperation is particularly essential when the relationships between strategic, tactic, and operative processes of an organization are explored. In general, governors and actors ensure the context-sensitive processing of information, supported by facilitators. They also try to capture complex situations, e.g., looking why performance parameters cannot be met, and consult (domain) experts when required.

5.2 S-BPM Activity Bundles

In S-BPM several bundles of activities can be identified (cf. [3]), in line with existing BPM approaches (cf. [2]). They are organized according to phases that need to be performed when organizations shift to process-centered value chains or want to implement business processes. The activities range from analysis to running and monitoring.

Analysis. S-BPM models traditionally play a crucial role for analysis, as they are recognized as valid start documentation of BPM projects. They can be created in the course of analysis, if not already existing models are revisited at this stage of development. In any case, the situation of an organization 'as it is' needs to be documented as a result of the analysis process, otherwise the origin or trigger of organizational development gets lost, and changes cannot be checked against a well defined start state of a change or development process. Facilitators, governors, and actors need to collaborate to produce sound analyses. They might be supported by method specialists or domain experts to develop deep understanding of the underlying processes.

Modeling. Traditionally, modeling needs to be considered as a separate bundle of activities in BPM, since envisioned processes are specified at this stage of organizational development or change management. Of particular interest in modeling is reducing complex relationships when operating processes, while keeping the specification of business processes coherent. S-BPM allows achieving both:

- *Complexity is reduced* by focusing on subject behavior: Internal behavior is encapsulated in subject-specific models. S-BPM modeling also leads to task- and business-relevant interactions between subjects - the adjusted behavior specifications represent the overall organization of work. In this way, only domain-relevant subjects are considered in the course of modeling. Their structure (representing the organization of work) and internal behavior are strictly separated, while being intertwined in a coherent way: For each subject in the S-BPM Interaction Diagram an S-BPM Behavior Diagram needs to be specified. Finally, for each subject an internal behavior is provided. Its interfaces correspond to the message relationships to those subjects the subject needs to interact with in the course of task accomplishment as given in the S-BPM Interaction Diagram.
- *Coherence is ensured* twofold: First, as it is relevant on the organizational level, interaction patterns need to be complete per se - a sender requires a receiver (and vice versa). Consequently, no incomplete interactions to that respect are allowed. Secondly, being relevant for accomplishing tasks and achieving outcome: Each received message triggers functional behavior of a subject, and leads to further interaction delivering work results (up to the customer), as represented in S-BPM behavior and interaction diagrams.

Apparently, S-BPM models are the core element in the modeling phase. In S-BPM the organizational and subject-specific level, and their interfaces are addressed in a consistent way. An organization is represented in terms of interacting subjects specified in the S-BPM Interaction Diagram. Outcome is generated through the exchange of business objects that are processed by functions. Functions are performed by the involved subjects, and are specified in the S-BPM Behavior Diagram.

In this way, S-BPM captures all essential aspects of BPM, namely the Who, the What, the How, and the When. However, it is the communication-oriented way of specifying organizational and stakeholder behavior ensuring coherence and reducing complexity in change management.

Stakeholders are encouraged by governors and facilitators to participate as active modelers throughout modeling. They might be supported through domain experts or (S-)BPM specialists.

Validation. The validation challenges the effectiveness of process models. It is checked whether a process allows producing expected results. As such, a reality check is performed to ensure the expected process performance through models. In this bundle of activities mostly the actors and method specialists are involved, as they need to validate communication and functional procedures with respect to the quality of the results. Facilitators and governors guide them and monitor the process.

Optimization. Once a process has been validated, it can be optimized to certain criteria, such as trying to achieve short information paths. In that phase the efficiency of a modeled process is checked. Resource-specific aspects such as time and material consumption are investigated, and might lead to significant changes of models. Again, actors and experts play a leading role in that phase of S-BPM. Here, the approach has the unique benefit that subject-specific behavior, e.g., processing a specific type of material, is primarily encapsulated, but still embodied in its processing context - all interfaces in terms of exchanged information and business objects are represented explicitly.

Embodying. Analysis, modeling, and validation occur decoupled from the operational processes going on in an organization. In order to feed the results back to the running business, the process models have to be implemented in the organization. Here, the governors and facilitators take leading roles, as mainly the structure, infrastructure, and strategy of an organization are affected by that task. They might consult actors and method specialists. The implementation is performed on the level of organizing work, and on the level of infrastructure, e.g. IT systems. It may require domain experts to facilitate this process.

Running and Monitoring. Once in operation, business processes need to be monitored. In doing so, data are recorded and observations are collected that might trigger further S-BPM analyses and modeling activities. Governors guide this process, eventually consulting actors, experts, or asking facilitators for additional guidance. In S-BPM each actor can track models and trace behavior on the subject and organization layer, as the implemented workflow is a 1:1 mapping of the S-BPM Interaction and Behavior Diagrams to functional software components. Of particular interest at this development stage are meeting expectations, e.g., in terms of removing work hindrances or meeting performance requirements, and recognizing the initialization of further S-BPM activities.

5.3 Capturing the Dynamics of Organizational Development

S-BPM is considered a multi-layered change process. Particular bundles of activities or iterations of several bundles enable the emergence of novel organizational behavior, becoming manifest in the various levels of organizational development. Each level corresponds to a certain level of organizational maturity, and can be achieved either in a linear or a non-linear sequence of S-BPM activity bundles, as indicated in Figure 14.

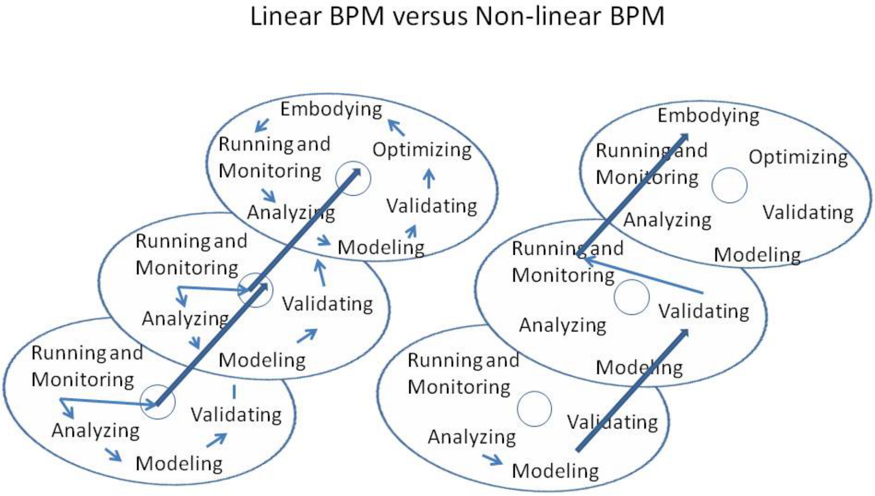


Fig. 14. Patterns of organizational development driven by S-BPM

Linear development (left part of the figure) corresponds to traditional life cycle approaches to BPM: In order to complete a phase in S-BPM, each activity has to be executed, and needs to be completed at least one time before entering the next life cycle (i.e., the next level in development), even when there are cyclic activities within each life cycle, such as modeling and validating models several times. The transition to the next BPM step is traditionally defined by reaching a dedicated bundle of activities, mainly running and monitoring. It allows observing running a business after modeling and embodying processes into the operation, and before analyzing the effect of implemented process changes. It corresponds to entering already the next BPM cycle, as indicated when following the bold directed link to the upper level in the figure.

In the *non-linear S-BPM approach* (right part of the figure) reaching the next step of organizational development is characterized by being able to switch to a higher stage of development (displayed as upper layer) from each of the activities, as indicated in the figure through the bold directed arcs. The most typical example is changing individual functional behavior while keeping the interaction interface to other subjects. It allows improving the individual organization of work on the fly.

However, its effects become evident on the organizational level through monitoring the concerned subject's behavior in its operative context. Since this emergence of organizational behavior resulting from individual functional behavior modification can be driven by several subjects, the results need to be evaluated (monitoring and analyzing) on another level of organizational development than the one where the changes actually occurred.

The more an execution engine is intertwined with the activities of the life cycle the more direct effects of changes can be experienced and the more likely stakeholder changes lead to the next level of organizational development. It accelerates organizational development.

When handling the S-BPM life cycle in a non-linear way, modeling has to be considered one of the core activities, as models may serve as focal point for improvements or for changes of the communication behavior before becoming effective on the operational level. Due to the coherent decomposition of processes and the resulting behavior management, a step closer to organizational reality can be made.

6 Conclusion

Organizations are increasingly forced to restructure their business processes in a flexible way during operation. It requires stakeholders to take responsibility for organizational developments. Traditionally, only few members are skilled in specifying and developing business processes. When using subject-oriented BPM models they can work with natural language constructs (subject, predicate, object) and e-mail-like communication patterns between actors or systems when describing business processes. In this way, individual members of an organization are able to contribute to coherent and intelligible process specifications. In addition, S-BPM models can be processed without transformation, allowing hands-on experience of business processes.

Subject-oriented representation schemes recognize actors or systems as starting point for modeling, regardless of how they evolve. In case they are constructed from scratch they are identified successively, according to the flow of work, still taking into account subjects as part of standard sentence semantics, namely adding operations and business objects in the course of modelling. In case S-BPM models are constructed by restriction, a general communication pattern is successively aligned to the required flow of information and material between the actors or systems necessary to complete a specific process.

Using subjects, stakeholders avoid conveying information reduced either to content or functional business logic. It also ensures coherence, as both, the flow of control, and the addressed data can be kept in their respective context throughout modeling and execution business processes. Consequently, stakeholders and developers should experience less misunderstandings and conflicts in industrial practice. This benefit becomes essential for networked organizations striving for interoperability, as social interaction, cooperation and collaboration aspects have to be reflected by modelling techniques.

References

1. Acosta, A., Leon, Y.J., Conrad, C.R., Malave, C.O.: Global engineering: Design, decision making and communication. CRC Press, London (2009)
2. Becker, J., Mathas, C., Winkelmann, A.: Geschäftsprozessmanagement. Springer, Berlin (2009)
3. Fleischmann, A., Schmidt, W., Stary, C., Obermeier, S., Börger, E.: Subjektorientiertes Geschäftsprozessmanagement. Hanser, München (2011)
4. Heftberger, S., Stary, C.: Partizipatives organisationales Lernen – Ein prozessorientierter Ansatz. DUV, Wiesbaden (2004)
5. Herrmann, T., Hoffmann, M., Kunau, G., Loser, K.U.: A modelling method for the development of groupware applications as socio-technical systems. Behavior & Information Technology 23/2, 119–135 (2004)
6. Pinker, S.: The Stuff of thought: Language as a window into human nature. Allen Lane, London (2007)
7. Scheer, A.-W.: ARIS - Modellierungsmethoden, Metamodelle, Anwendungen, 4th edn. Springer, Berlin (2001)
8. Schmidt, W., Fleischmann, A., Gilbert, O.: Subjektorientiertes Geschäftsprozessmanagement. In: HMD – Handbuch der Wirtschaftsinformatik, Heft 266, pp. 52–62 (2009)